FSB Measurement, Analysis and Reporting Standard

Flight Software Branch – Code 582

Version 1.0 – 12/14/05

582-2005-008

Goddard Space Flight Center
Greenbelt, Maryland

National Aeronautics and
Space Administration

FORWARD AND UPDATE HISTORY

This document describes the processes for planning, analysis, reporting, and archiving of Flight Software Branch measurement data.

Version	Date	Description	Affected Pages
0.1	09/03/03	Layout changes, new cover page	All
0.2	12/02/05	Revisions in response to pre-assessment	All
1.0	12/14/05	Formatting changes prior to baselining, BSR slide examples replaced with most recent versions.	All

CONTENTS

1 INTRODUCTION

This standard describes how Flight Software teams are to carry out required measurement activities. It outlines the responsibilities of software teams, and the support that is provided by the FSB Measurement Team.

The remainder of this section describes the allocation of responsibilities between teams. Section 2 describes how measurement is planned for at the beginning of a project, Section 3 describes how data are collected and stored throughout the project. Together, Sections 2 and 3 provide an overview of how to plan and carry out all measurement requirements. Section 4 provides the detailed procedures for how data are analyzed and reported throughout the project.

1.1 FLIGHT SOFTWARE DEVELOPMENT TEAMS

The flight software development teams have the following measurement responsibilities:

- Manage project using measures. Measurement is used here to assess whether a software project is proceeding according to plan, and to inform any necessary changes if a project deviates from plan.

- Meet software measurement requirements, as defined in NPR 7150.2, and requirements of assessment models such as CMMI or ISO. The approach described in this standard is consistent with these requirements; so following this standard will ensure that these requirements are being met.

1.2 FLIGHT SOFTWARE BRANCH MEASUREMENT TEAM

The Flight Software Branch Measurement Team has been formed to provide support for all measurement activities in the Flight Software Branch. This support includes:

- ***Logistical support for FSW projects*** in the collection, storage, analysis and reporting of measurements needed to manage their projects and to meet NPR 7150.2 and CMMI measurement requirements. This includes both developing a standard tool set for the Branch and supporting each team’s measurement operations. The ISD Measurement Team may provide resources to help with the tool development.

- ***Performing cross project modeling and analysis*** to create more accurate branch approaches to project planning and project monitoring and control. Experts from the ISD Measurement Team may support this work.

- ***Provide data to organizational measurement programs*** at the Division, Center or Agency level. These data are a subset of data already collected for software projects; however it will be the FSB Measurement Team’s responsibility, not the software team’s, to report this data upwards.

2 PLANNING FOR MEASUREMENT

This section describes the planning and setup work that needs to be done to start measurement for a Flight Software project. This consists of documenting the measurement approach in the Product Plan (Section 2.1) and setting up the data collection and storage procedures (Section 2.2). The former is carried out by the Product Development Lead as part of writing the Product Plan; the latter is carried out together by the PDL and a representative from the FSB Measurement Team.

Once the work described in this section is complete, the team is ready to perform all the required measurement data collection storage, analysis and reporting.

2.1 DOCUMENTING A MEASUREMENT PLAN

The measurement plan is documented in Section 3.10 of a Flight Software Product Plan, as shown in the FSW Product Plan Template. The Product Plan Template section must be tailored for a specific project by inserting project specific language into the introductory text, and creating a project-specific version of Table A-1, found in Appendix A of this standard. The former may be as easy as filling in the Project name at specific locations in the template; tailoring the table involves a greater number of decisions. They are presented below in rough order of likelihood.

> *Identify specific tools used by this particular team*. This is done by identifying the tools used to collect and store raw data in the "Measurement Mechanism / Artifact" column. Some of the tools, for example the Point Counting spreadsheet, are branch standards. Others, for example the mechanism for tracking action items, can be chosen from a list of options. In the latter case, the selected option must be documented. **It is required that the product plan reference the specific tools used for this particular project**.
>
> A*djust analysis as needed*. Generally one should use the triggers for action pre-defined in Appendix A. However, a Product Development Lead may change these if there is sufficient difference from the FSB norm to make this reasonable. **No change to the analyses is required, but it is permitted with justification**.
>
> *Add metrics*. This is done if there is information desired by the Product Development Lead that is not covered in the standard measures. As an example, the FSB does not require analysis of how long DCRs remain open. However, a project using a branch reuse library might want to track how long DCRs against the reuse library take to resolve. To add metrics requires adding a row in the appropriate section of the table. To continue the reuse library example, one would add cells for Measurement Objective, Analysis Summary, Measure(s), Measurement Mechanism/Artifact, Collected By, and Collection Frequency cells under the Software Quality Measurement Area. The Product Development Lead is then responsible for writing contents for these cells. **There is no requirement to add any metrics to the standard set, but it is permitted**.

Upon completing this table, the next step is to work with the FSB Measurement Team to set up data collection, and storage procedures.

2.2 SETTING UP MEASUREMENT PROCEDURES

As described in Section 1.2, the FSB Measurement Team provides logistic support to FSW software teams. The FSB Measurement Team is responsible for collecting the data, storing the data, and generating draft Branch Status Report slides for the software team to use. The software team's responsibilities are to provide the data needed to generate these spreadsheets and draft slides, and to edit the slides to add their analysis of the status.

The monthly metrics data are collected in the following spreadsheets:

- FSW Metrics Spreadsheet
- FSW Status Spreadsheet
- FSW Point Counting Spreadsheet
- FSW Staffing Spreadsheet.

At the start of a project, the following data need to be collected to set up the spreadsheets, as well as to satisfy other measurement requirements:

Software Characteristics: These data, required by NPR 7150.2, include a listing the major subsystems. This information is used to set up spreadsheets for data that is reported on a per subsystem basis, such as the number of open and closed DCRs by severity stored in the Metrics Spreadsheet.

Initial Plans: These plans included a staffing plan, the planned number of progress points to be attained each month, and the planned number of requirements to be met by each build as delivered to the test team. All of these plans are detailed enough to show allocations of staff, progress and requirements to each subsystem.

The software team and the FSB Measurement Team need to meet at the beginning of the project to collect the information needed to set up the spreadsheets, and to agree on the logistics for collecting data from the team and returning the draft BSR slides.

In addition to the monthly data, the following measures need to be collected, stored, analyzed and reported:

- Initial cost estimates and yearly POP updates, reported after the annual POP exercise.
- Basis of Estimate (BOE) information associated with the cost estimates.
- Critical performance measures such as CPU, bus, telemetry bandwidth and memory utilization, reported at major milestone reviews.

3 DATA COLLECTION AND STORAGE PROCEDURES

This section outlines the measurement activities of a software team and the logistical support provided by the FSB Measurement Team throughout the project. There are three points at which data are collected and stored. One collects some of the data either by directly entering it in the spreadsheet or generating a report from a tool and using the output as input to Excel or Power Point. The procedures for each of these are outlined below. Each project writes a detailed collection and storage procedure that describes file naming conventions, directory locations for spreadsheets and tools, archival procedures, and detailed instructions for each spreadsheet.

Monthly Branch Status Reports

- The PDL or delegated team member collects metrics to be provided to the FSB Measurement Team.

- The FSB Measurement Team enters the data into the Metrics Spreadsheet, Staffing Spreadsheet, Point Counting Spreadsheet and Status Spreadsheet. NOTE: A software team may agree to take responsibility for directly entering some or all of this data, but that is not required. Whatever alternative is agreed to must be documented in the collection and storage plan.

- The PDL and team leads add analysis to the charts generated in the spreadsheets. This analysis is presented to senior Branch management at the Branch Status Review.

Milestone Reviews

- The software team will provide the metrics presented at major milestone reviews to the FSB Measurement Team.

- The FSB Measurement Team will enter the data into the Branch Measurement Repository (currently under the two “Milestone Data” tabs in the FSW Status Spreadsheet), and forward the required subset to the ISD Measurement Repository.

Annual POP Exercises

- TBS

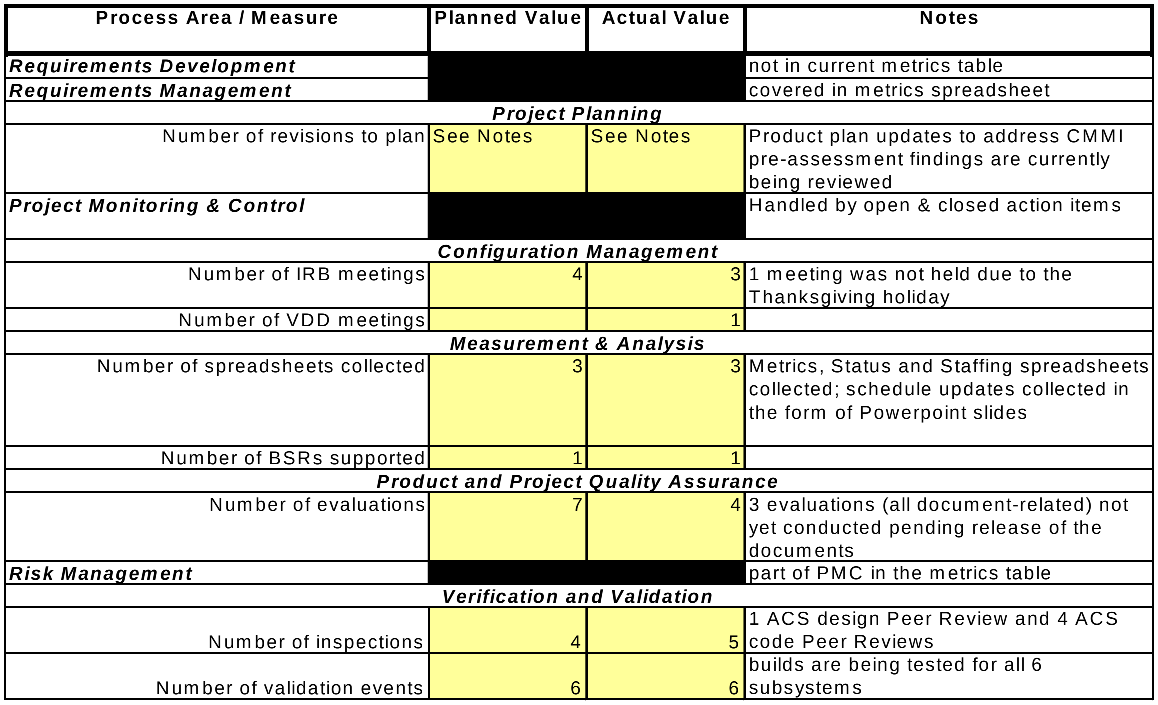
4 ANALYSIS AND REPORTING PROCEDURES

This section describes the analysis performed by an FSB team's Product Development Lead and supporting team leads in preparation for review meetings such as the monthly BSR or milestone reviews. It defines a single analysis procedure for each Measurement Area defined in Table A-1, presenting the objectives being attained, the measures being used, and the analyses to be performed.

The objectives can be classified as being related to the product being developed or to the development processes being used. The former are used to assess whether or not the software will be ready on time, on budget, and correctly satisfying all its requirements. The latter provides evidence that the Branch-defined processes are being carried out as intended. These objectives are documented in Section 3.10 of each project's Product Plan.

The product measures are collected in the spreadsheets described in Sections 2 & 3. Each procedure shows a slice of the spreadsheet used to collect the measures; the yellow areas indicate where data is entered, the remaining values are calculated from this data. In most cases, not all the subsystems are presented, in the interest of saving space. Similarly, many of the spreadsheets have room for 18 months of data, this amount is truncated to fit into a portrait presentation without reducing the table to an eye chart.

The process measures fall into two categories. Some of them have their own tabs to enter data; for example the action items have their own "AI Data" tab in the Metrics Spreadsheet. In some cases, most notably for requirements management, the same data can be analyzed as both process and product metrics. In the case where a separate tab does not exist, the process metrics are collected under the "Proc. Data" tab of the FSB Metrics Spreadsheet, as shown in Figure 4-1 below. Each of these data items refer to counts for the month, and can be filled in by looking at the planned and actual completion of these events on the team's Gantt charts.



Process Area / Measure	Planned Value	Actual Value	Notes
Requirements Development			not in current metrics table
Requirements Management			covered in metrics spreadsheet
Project Planning			
Number of revisions to plan	See Notes	See Notes	Product plan updates to address CMMI pre-assessment findings are currently being reviewed
Project Monitoring & Control			Handled by open & closed action items
Configuration Management			
Number of IRB meetings	4	3	1 meeting was not held due to the Thanksgiving holiday
Number of VDD meetings		1	
Measurement & Analysis			
Number of spreadsheets collected	3	3	Metrics, Status and Staffing spreadsheets collected; schedule updates collected in the form of Powerpoint slides
Number of BSRs supported	1	1	
Product and Project Quality Assurance			
Number of evaluations	7	4	3 evaluations (all document-related) not yet conducted pending release of the documents
Risk Management			part of PMC in the metrics table
Verification and Validation			
Number of inspections	4	5	1 ACS design Peer Review and 4 ACS code Peer Reviews
Number of validation events	6	6	builds are being tested for all 6 subsystems

Figure 4-1 Process Monitoring Data.

This worksheet either provides space to enter the appropriate data or references where the data for a process area can be found. The actual analysis and reporting is discussed in the analysis procedure for each process area.

The analysis section of each procedure shows how to evaluate the data in these tables, or in plots generated from these tables. The text describes how to analyze both tables and plots, including how to identify areas of concern from the data. The text also describes how to assess the impact of issues, and identifies potential corrective actions for consideration. Sometimes the action is as simple as raising the concern at a review, other times it may indicate more significant changes such as shifting staff assignments, or even adding staff.

4.1 PROJECT MONITORING AND CONTROL PROCEDURE

4.1.1 Objectives

The objectives for Project Monitoring and Control (PMC) fall into two broad categories. The first is to monitor software progress to determine that the team is on track to deliver the required functionality on time and within budget or not. The second is to assure that the Project Monitoring and Control process is being carried out properly.

The objectives for software progress are to:

- Monitor schedule progress to ensure milestones can be met.
- Ensure product progress is adequate to ensure completion by scheduled date.
- Monitor effort and cost to ensure completion within budget.
- Ensure the management of project risks.

The objectives for process monitoring are to:

- Ensure project issues are identified and resolved in a timely manner.
- Ensure adequacy of resources for Project Monitoring and Control.

4.1.2 Measures

Measures for software progress

The measures for software progress are:

- Planned and actual event dates. These include both milestones and process events.
- Planned and actual progress tracking points
- Planned and actual effort for civil servants and contractors
- Number of added, modified and retired risks by severity.

The template for entering schedule information is currently under development; thus collecting milestone and event dates is defined as part of each project's plan. Each project must be able to create Gantt charts of the sort shown in Figure 4.1-5. A future revision of this standard will define a common approach for collecting this data.

PLACEHOLDER FOR FUTURE VERSIONS OF DOCUMENT

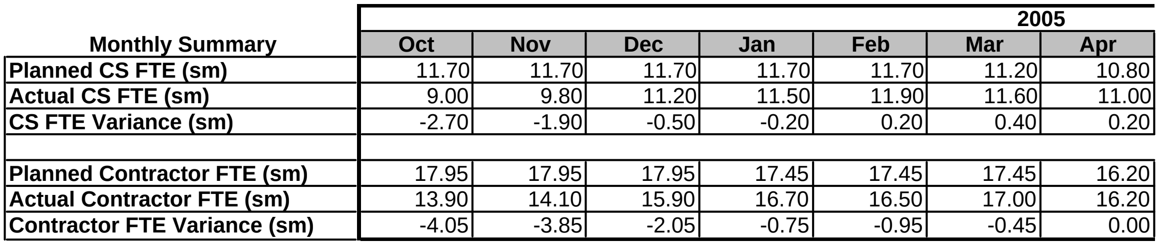
Figure 4.1-1 Milestone & Event Dates

Progress tracking points are tracked in the Point Counting Spreadsheet. The details of how to use this spreadsheet are contained in the template itself, under the “Help” tab and the ISD Point Counting Tool User’s Guide (http://software.gsfc.nasa.gov/toolsDetail.cfm?selTool=1.4.2.1). This spreadsheet is used to track the planned and actual completion of assignments at the lowest level of detail that is reasonable to track, and is set up to compute total planned and actual progress points from this detailed information. Figure 4.1-2 shows the cells of this spreadsheet that are computed from the more detailed information and plotted in figure 4.1-6. The frequency of updating the progress computations is under user control; nominally it should be updated weekly.

Plan	88	88	88	104	161	222	291
Actual	63	63	63	63	108	123	146
Baseline	615	615	615	615	615	615	615

Figure 4.1-2 Progress Points.

The effort and cost information is tracked using the FSB Staffing Spreadsheet. The documentation of this tool is embedded in the spreadsheet template itself, under the “User’s Guide” tab. The spreadsheet contains both cost data and staffing information broken down to the level of individual team members. Figure 4.1-3 shows how the data is rolled up to show the total planned and actual effort for both civil servants and contractors. This rollup is used to generate the plots in Figures 4.1-7 and 4.1-8.



							2005
Monthly Summary	**Oct**	**Nov**	**Dec**	**Jan**	**Feb**	**Mar**	**Apr**
Planned CS FTE (sm)	11.70	11.70	11.70	11.70	11.70	11.20	10.80
Actual CS FTE (sm)	9.00	9.80	11.20	11.50	11.90	11.60	11.00
CS FTE Variance (sm)	-2.70	-1.90	-0.50	-0.20	0.20	0.40	0.20
Planned Contractor FTE (sm)	17.95	17.95	17.95	17.45	17.45	17.45	16.20
Actual Contractor FTE (sm)	13.90	14.10	15.90	16.70	16.50	17.00	16.20
Contractor FTE Variance (sm)	-4.05	-3.85	-2.05	-0.75	-0.95	-0.45	0.00

Figure 4.1-3 Summary of Staffing Data

The data for risk is generated from the individual risks entered into the FSB Risk tool. One extracts these measures by generating reports from this tool, and using a screen-shot utility (e.g., the free utility found at http://www.wisdom-soft.com/products/screenhunter.htm) to generate the slide shown in Figure 4.1-9.

Measures for process monitoring

The measures used to analyze software functionality are:

- Number of open vs. closed action items
- Planned and actual effort for Management.

The action item data is collected in the FSW Metrics Spreadsheet, under the “AI Data” tab. Figure 4.1-4 shows this spreadsheet. All action items tracked via the software team’s action item tool are counted in this data and used to generate the plots in Figure 4.1-10.

	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Action Items Opened	3	6	8	12	15	15	15	16	13	
Action Items Closed	0	3	4	7	12	14	15	17	14	
Cum. Action Items Opened	3	9	17	29	44	59	74	90	103	#N/A
Cum. Action Items Closed	0	3	7	14	26	40	55	72	86	#N/A
Action Items Currently Open	3	6	10	15	18	19	19	18	17	#N/A

Figure 4.1-4 Action Item Data.

The effort data is entered into the staffing spreadsheet; table 4.1-12 is generated automatically from the staffing spreadsheet, is copied and pasted to the status spreadsheet, and analyzed there.

4.1.3 Analysis

Analysis for software progress

Software progress is analyzed by examining schedule, point counting, and staffing spreadsheets to assess the current health of a project, and using the reports generated by the risk tool to make the most important risks and the change in risks visible.

Figure 4.1-5 shows a Gantt chart with milestones. Gantt charts are used to provide an overview of milestones, and to show the schedule of critical events. The triangles indicate scheduled events, they are filled in when the event actually occurs. These events include technical, management and process events. A typical BSR has several such Gantt charts in it, organized according to the FSB standard WBS.

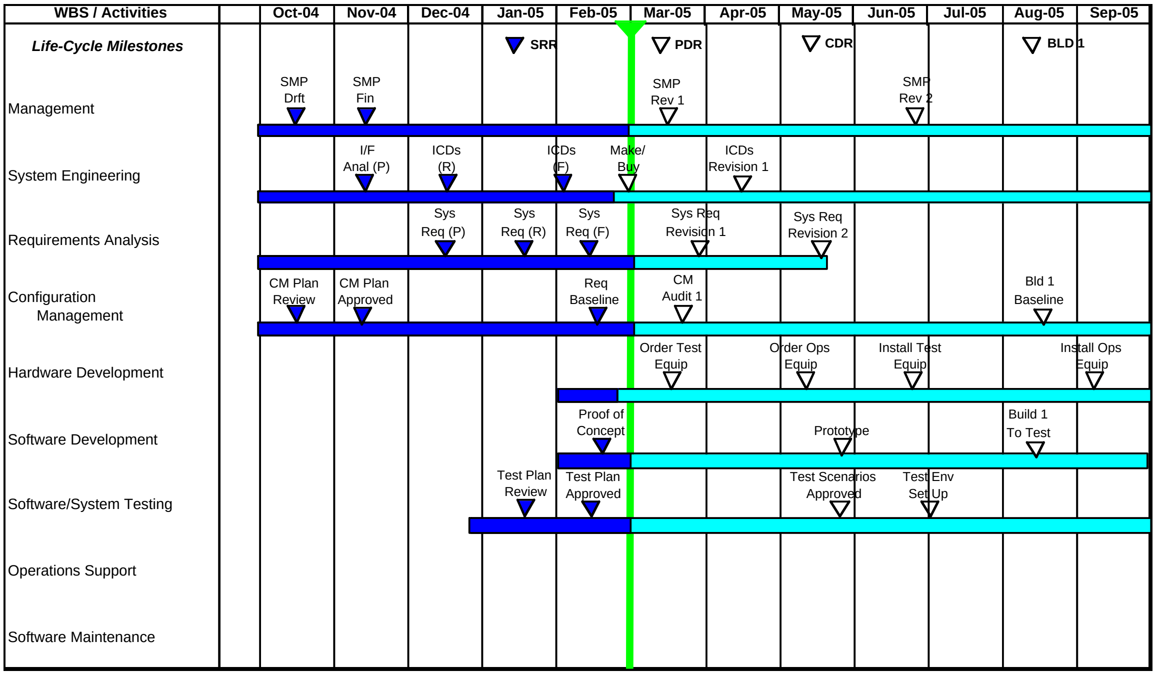


Figure 4.1-5 Gantt Chart for Schedule Progress

Figure 4.1-6 shows the progress tracking point status as generated from the Point Counting Spreadsheet. The blue line indicates the baseline for the subsystem and build in question; that is the total points assigned to the products being delivered. The green and red lines show the planned and actual number of points over time, respectively. This example does not show it, but one can use the baseline to see if the target for a build is fluctuating due to requirements changes or other issues.

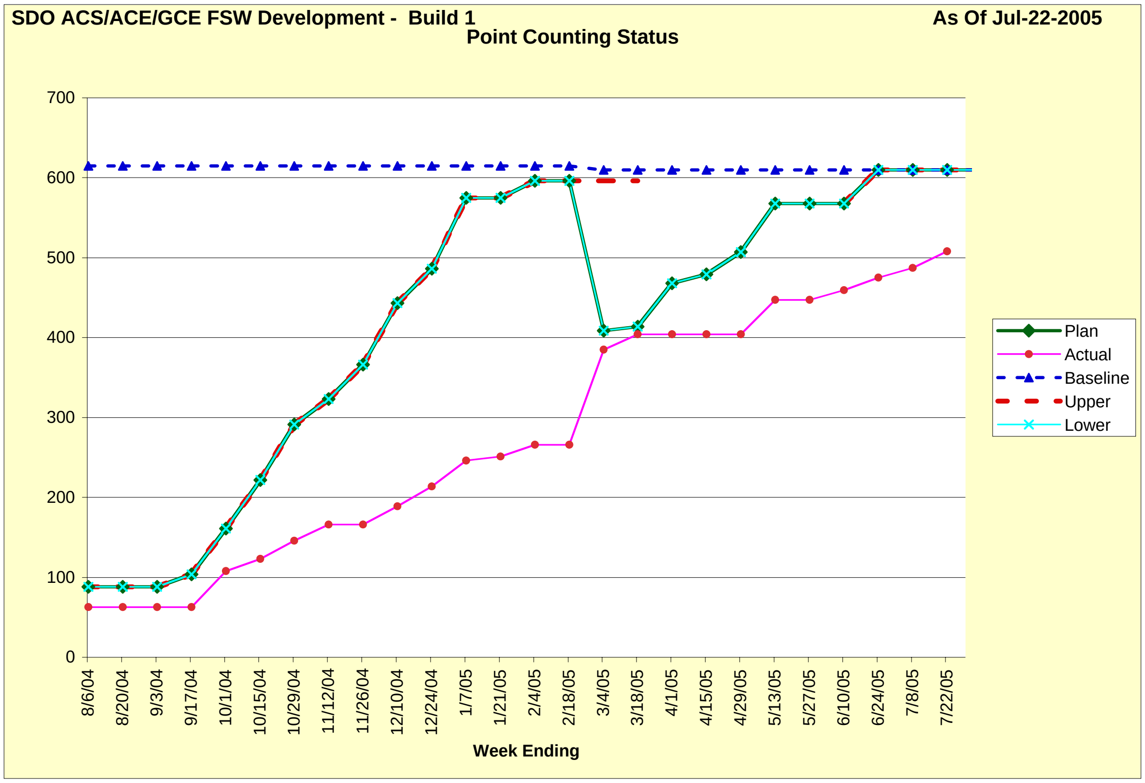


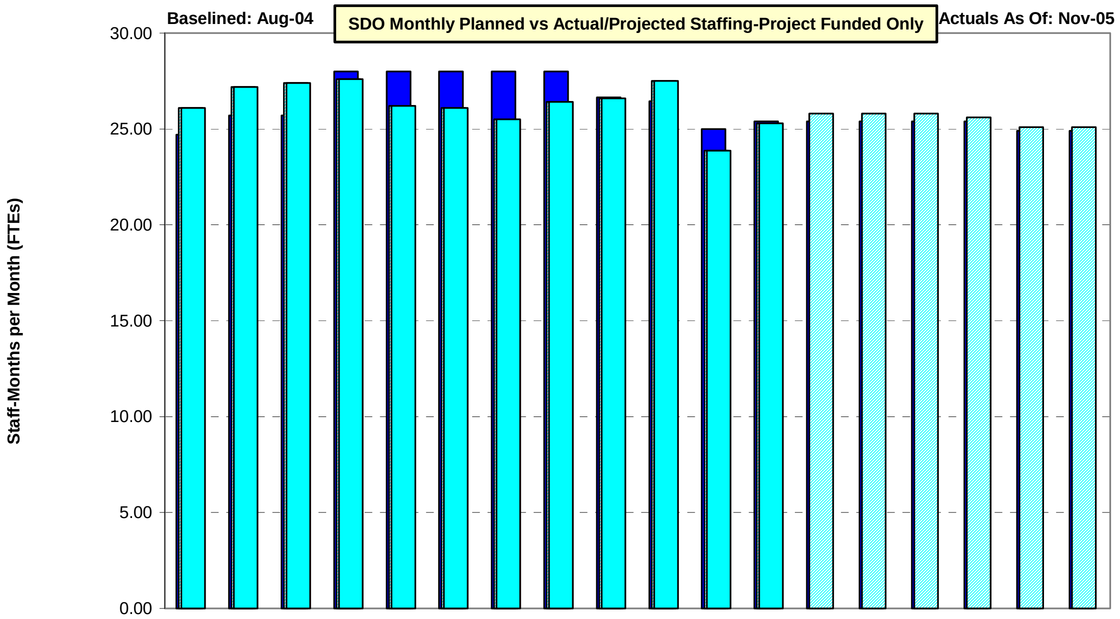
Figure 4.1-6 Progress Tracking Points

The data shown on these two graphs complement each other and should be analyzed together. If this analysis shows that the project is sufficiently far behind (nominally 10%), the corrective actions to consider include adding staff and budget, giving schedule relief, or moving requirements to fit resources available. In the latter case, capabilities can be moved to a later build, or in severe cases the final system functionality may need to be reduced. In the example shown above, the sharp decrease in the green (plan) line reflects an acknowledgement that the delivery will be later than planned.

The main reason to look at these data together is that at times one shifts resources to do other work while waiting for a delivery needed to make progress in the scheduled area. On the Gantt chart, one would only see missed events; the progress tracking would highlight the compensating work that is mitigating the issue. If this is the situation, the report would say that this mitigation would happening, and the risk posed by waiting for the delivery would be highlighted via the risk management tool and risk reporting (see below).

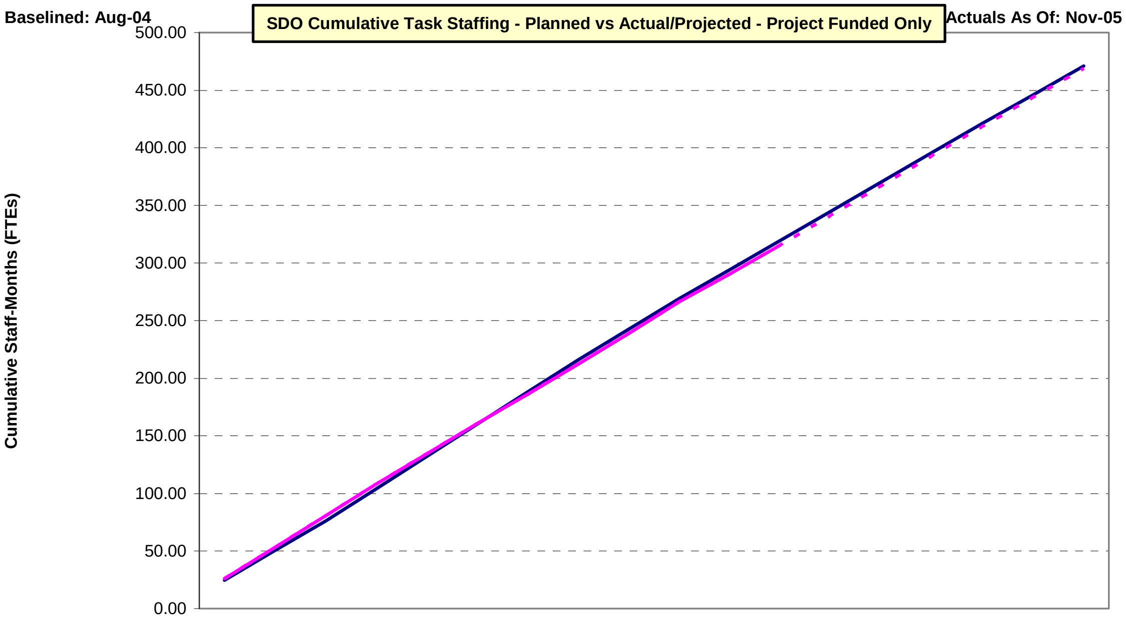
Looking a schedule and product progress only tells half the story. As an example, the delay shown in Figure 4.1-6 could be due to lower than expected productivity, or it could be due to understaffing. To determine which is the case, one would look at the staffing data.

Figures 4.1-7, 4.1.8, and 4.1-9 show three views of staffing data. Figure 4.1.7 shows a bar chart of planned and actual staffing for each month in an 18 month period. Figure 4.1-8 shows the cumulative staff for the same period. Figure 4.1.9 shows the cumulative staff for the lifetime of the project.



	Dec-04	Jan-05	Feb-05	Mar-05	Apr-05	May-05	Jun-05	Jul-05	Aug-05	Sep-05	Oct-05	Nov-05	Dec-05	Jan-06	Feb-06	Mar-06	Apr-06	May-06
■ Planned / Mo	24.70	25.70	25.70	28.00	28.00	28.00	28.00	28.00	26.65	26.45	25.00	25.40	25.40	25.40	25.40	25.40	24.90	24.90
▨ Projected / Mo	26.10	27.20	27.40	27.60	26.20	26.10	25.50	26.40	26.60	27.50	23.88	25.30	25.80	25.80	25.80	25.60	25.10	25.10
■ Actual / Mo	26.10	27.20	27.40	27.60	26.20	26.10	25.50	26.40	26.60	27.50	23.88	25.30						

Figure 4.1-7 Staffing Profile (18 month)



	Dec-04	Jan-05	Feb-05	Mar-05	Apr-05	May-05	Jun-05	Jul-05	Aug-05	Sep-05	Oct-05	Nov-05	Dec-05	Jan-06	Feb-06	Mar-06	Apr-06	May-06
Planned / Mo	24.70	50.40	76.10	104.1	132.1	160.1	188.1	216.1	242.7	269.2	294.2	319.6	345.0	370.4	395.8	421.2	446.1	471.0
Projected / Mo	26.10	53.30	80.70	108.3	134.5	160.6	186.1	212.5	239.1	266.6	290.4	315.7	341.5	367.3	393.1	418.7	443.8	468.9
Actual / Mo	26.10	53.30	80.70	108.3	134.5	160.6	186.1	212.5	239.1	266.6	290.4	315.7						

Figure 4.1.8 Staffing Profile (18-month, cumulative)

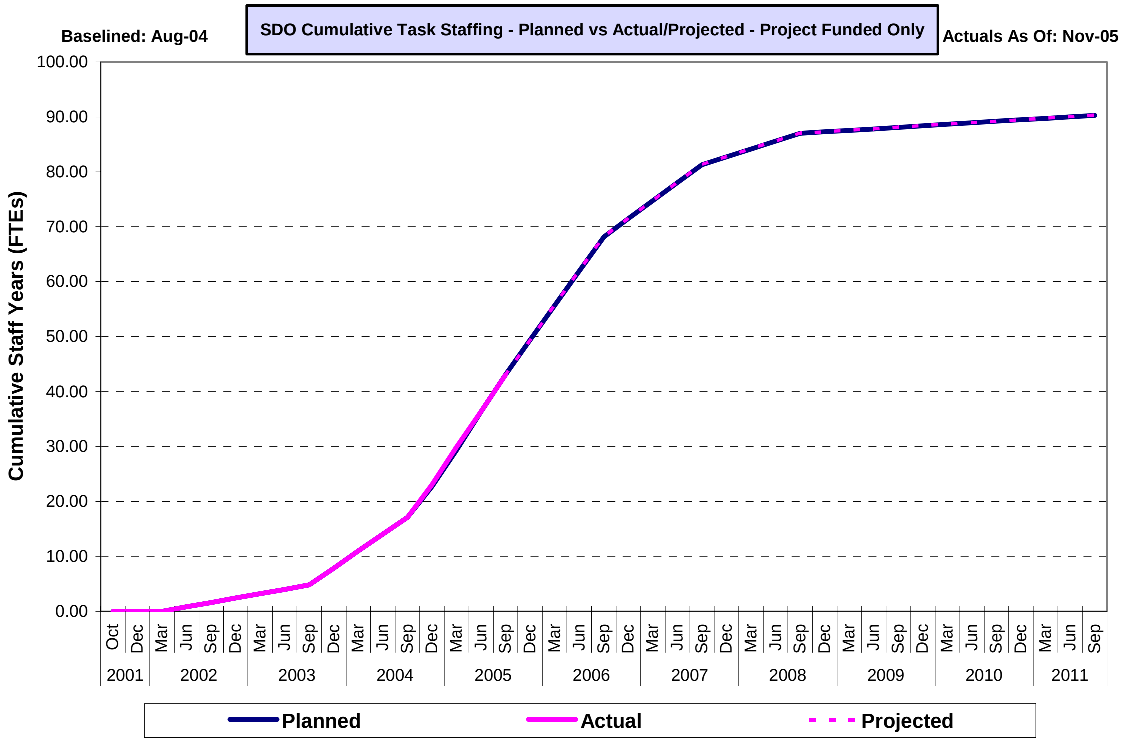


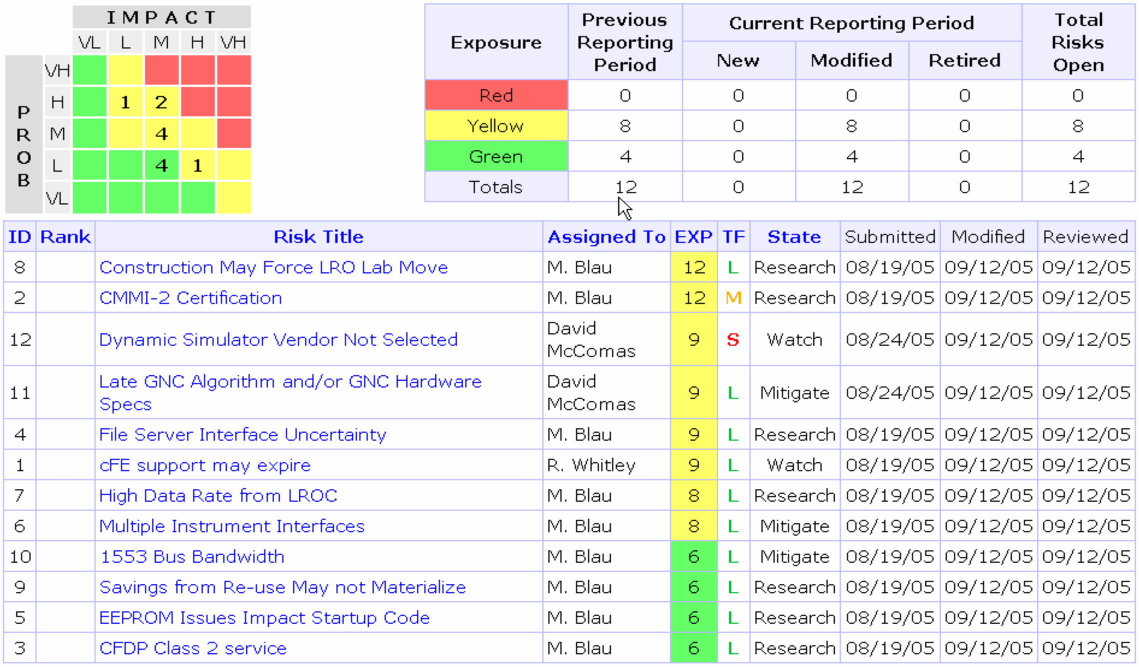
Figure 4.1-9 Staffing Profile (full project, cumulative)

In this example, Figure 4.1-7 shows the staff lagging early, then reaching planned levels. In this case, the cumulative version of these plots would show the actual curve first falling behind, the paralleling the plan; one would be remaining behind, and unless the team were more productive than the usual FSW team there would be a schedule impact.

As in schedule and product progress, the nominal threshold for taking corrective action is falling behind by 10%. This example has shown an analysis where schedule and progress points were analyzed first; it is equally possible to start with staffing and then to look at schedule and product progress. When all three types of data are examined together, one can get an accurate picture of where a project stands. Many managers already do this intuitively, putting this on a formal basis ensures that everyone will raise issues early enough to act before a situation is out of hand.

In some cases, you will see a problem without seeing an obvious cause such as understaffing or an increasing number of progress tracking points in a baseline. In this case, the thing to do is to correlate analysis of other data with this PMC data. As an example, lack of progress on a build may be due to a large number of requirements changes or TBDs, rather than understaffing. Most of these items are covered in the "impact and corrective actions" sections of subsequent analysis procedures.

Figure 4.1-10 shows the main risk slide in the BSR template, as generated from the FSB Risk Tool. It displays a count of risks organized by exposure (product of likelihood and impact), both in the matrix on the left and the table on the right. The key measure called out is the number of risks added, modified and retired, as shown on the right side. Finally, the top risks are listed at the bottom of the slide. In this case the entire list is shown, as the list is still pretty short.



PROB	IMPACT				
	VL	L	M	H	VH
VH	🟩	🟨	🟥	🟥	🟥
H	🟩	🟨 1	🟨 2	🟥	🟥
M	🟩	🟨	🟨 4	🟨	🟥
L	🟩	🟩	🟩 4	🟨 1	🟨
VL	🟩	🟩	🟩	🟩	🟨

Exposure	Previous Reporting Period	Current Reporting Period			Total Risks Open
		New	Modified	Retired	
Red	0	0	0	0	0
Yellow	8	0	8	0	8
Green	4	0	4	0	4
Totals	12	0	12	0	12

ID	Rank	Risk Title	Assigned To	EXP	TF	State	Submitted	Modified	Reviewed
8		Construction May Force LRO Lab Move	M. Blau	12	L	Research	08/19/05	09/12/05	09/12/05
2		CMMI-2 Certification	M. Blau	12	M	Research	08/19/05	09/12/05	09/12/05
12		Dynamic Simulator Vendor Not Selected	David McComas	9	S	Watch	08/24/05	09/12/05	09/12/05
11		Late GNC Algorithm and/or GNC Hardware Specs	David McComas	9	L	Mitigate	08/24/05	09/12/05	09/12/05
4		File Server Interface Uncertainty	M. Blau	9	L	Research	08/19/05	09/12/05	09/12/05
1		cFE support may expire	R. Whitley	9	L	Watch	08/19/05	09/12/05	09/12/05
7		High Data Rate from LROC	M. Blau	8	L	Research	08/19/05	09/12/05	09/12/05
6		Multiple Instrument Interfaces	M. Blau	8	L	Mitigate	08/19/05	09/12/05	09/12/05
10		1553 Bus Bandwidth	M. Blau	6	L	Mitigate	08/19/05	09/12/05	09/12/05
9		Savings from Re-use May not Materialize	M. Blau	6	L	Research	08/19/05	09/12/05	09/12/05
5		EEPROM Issues Impact Startup Code	M. Blau	6	L	Research	08/19/05	09/12/05	09/12/05
3		CFDP Class 2 service	M. Blau	6	L	Research	08/19/05	09/12/05	09/12/05

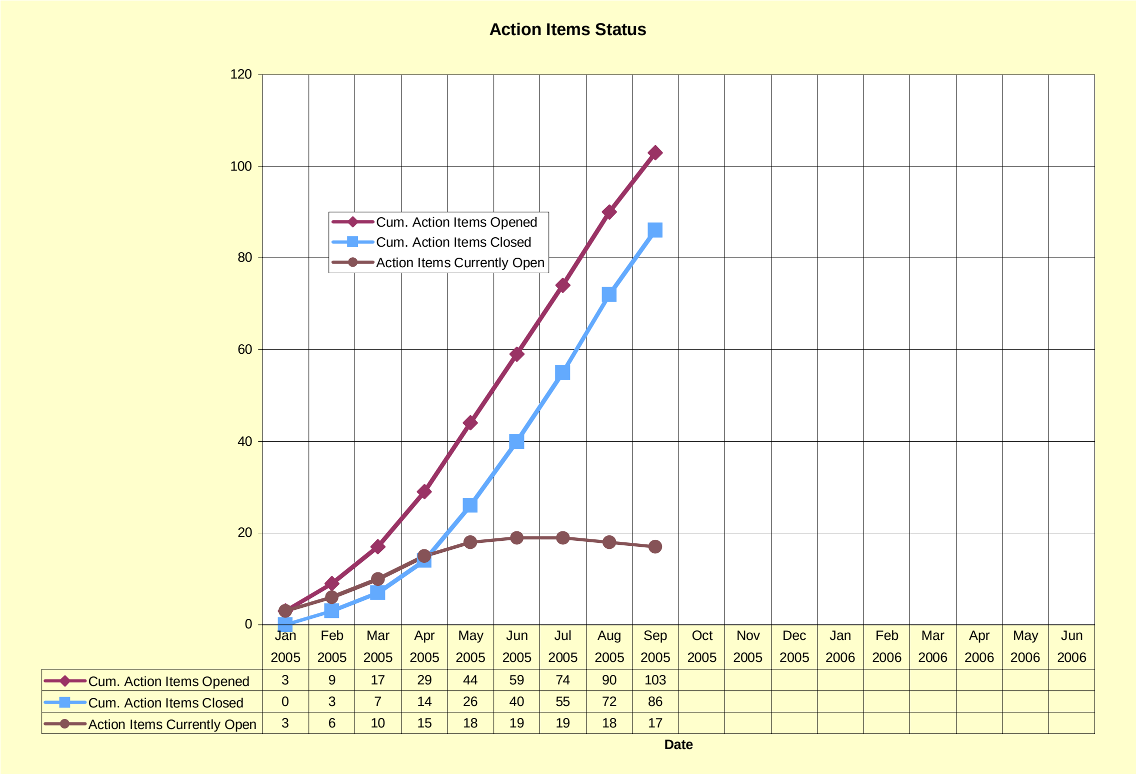
Figure 4.1-10 Risk report

There are two pieces of analysis of this data. The first is a form of process monitoring; if the new, modified and retired risks in the table on the right are all set to zero, then one has to ask if the project has really been monitoring them carefully enough. They may very well be doing so, but unchanging data is a good cue to double check.

The second analysis is the reporting on individual risks. Hard as it is to credit, this team believes CMMI poses a significant risk, and has decided it bears further research. The state of and results of this research need to be presented at each review until the analysis points to a change in the risk; whether for the better or for the worse. One should also be on the alert for short time frame risks, ones that need to be resolved quickly. In this example, the selection of a dynamics simulator vendor needs to be made quickly, and if it isn't the risk state will change from watch to something more serious.

Analysis for process monitoring

Figure 4.1-11 shows the plot of action item status. The red and blue curves show the cumulative number of items opened and closed, respectively; the lower curve shows the number currently open as a function of time.

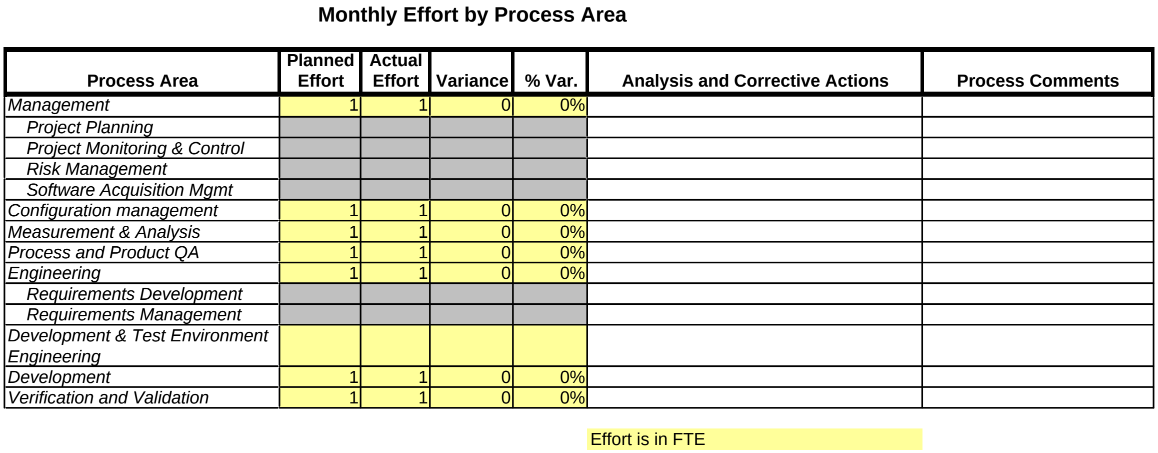


	Jan 2005	Feb 2005	Mar 2005	Apr 2005	May 2005	Jun 2005	Jul 2005	Aug 2005	Sep 2005	Oct 2005	Nov 2005	Dec 2005	Jan 2006	Feb 2006	Mar 2006	Apr 2006	May 2006	Jun 2006
Cum. Action Items Opened	3	9	17	29	44	59	74	90	103									
Cum. Action Items Closed	0	3	7	14	26	40	55	72	86									
Action Items Currently Open	3	6	10	15	18	19	19	18	17									

Figure 4.1-11. Action Item Status Plot.

This data is used to ensure that project monitoring and control is being carried out as planned. The essence of project monitoring and control is to monitor a project against its plan, and to take corrective action when it deviates from plan. Demonstrating that action items are indeed being identified and closed provides evidence that the project is being monitored and controlled. In this case, the number of open action items is remaining stable, and is at a reasonably low number. If the effort spent on monitoring and controlling the projects is not excessive in the PDL's judgment, the Project Monitoring and Control process area would be reported as "Green" in the appropriate row of Figure 4.1-12.

Figure 4.1-12 shows the Effort by Process worksheet, which is found under the "Effort by Process" tab of the FSW Status Spreadsheet. The corresponding analysis is discussed in the procedure associated with each process area, with the results being entered into the appropriate row of this worksheet and reported on a single chart for process monitoring at the BSR.



Monthly Effort by Process Area

Process Area	Planned Effort	Actual Effort	Variance	% Var.	Analysis and Corrective Actions	Process Comments
Management	1	1	0	0%		
Project Planning						
Project Monitoring & Control						
Risk Management						
Software Acquisition Mgmt						
Configuration management	1	1	0	0%		
Measurement & Analysis	1	1	0	0%		
Process and Product QA	1	1	0	0%		
Engineering	1	1	0	0%		
Requirements Development						
Requirements Management						
Development & Test Environment Engineering						
Development	1	1	0	0%		
Verification and Validation	1	1	0	0%		

Effort is in FTE

Figure 4.1-12 Process Monitoring Analysis

4.1.4 Impact and Corrective Action

If the behavior of effort, schedule and progress measures is nominal, there is no immediate impact on the project. Any changes in the most visible risks, whether the situation is improving or deteriorating, should be highlighted here to focus attention on icebergs looming ahead or icebergs successfully evaded.

If your performance is outside the nominal range in your favor, this is a good place to spread the credit, whether to your own team or exceptionally helpful to collaborating organizations.

In both the above cases, no corrective action is needed. In the unfortunate third case, you are outside the nominal range and behind the 8-ball. It doesn't matter what the root cause of the problem is, you are in a situation where the amount of work and the resources to do the work just don't match up. The corrective actions available are to

- Add resources, whether by adding people or by getting schedule relief. These resources can come from your own team's reserves or the issue can be raised to the Project for relief.

- De-scoping work by removing lower-priority requirements.

Another option is to prod people into working unpaid overtime. This is less than desirable, one of the above two options is preferable.

4.2 PROJECT PLANNING PROCEDURE

4.2.1 Objectives

The objectives for Project Planning are to ensure:

- That project planning is being performed as planned.

- Adequacy of resources for project planning

4.2.2 Measures

The measures for Project Planning include:

- Number of planned and actual revisions to budget and schedule.
- Planned and actual effort spent on management for the month.

The effort data is generated automatically in the Staffing Spreadsheet, on the "Effort by Process" tab. This data is copied to the Status Spreadsheet, where the analysis is performed (see Figure 4.1-11).

The count of planned and actual revisions is found in the FSW Metrics Spreadsheet, under the "Proc. Data" tab (see Figure 4-1).

4.2.3 Analysis

Nominally, re-estimation is done for the major milestone reviews and for annual budget exercises (POPs), and the plans should reflect this. If actual data for a month is within 25% of the plan and the re-estimates are carried out in time, then all is nominal and no corrective action is necessary.

If the actual effort is more than 25% away from the plan for a month, examine past data to determine if this is a persistent issue; if it is above the planned amount look further to see if there is project planning work not accounted for in your planned effort. If no corrective action is needed, simply report the reason for the one-time deviation.

If the actual effort is within this upper bound, but the re-estimates are not completed in a timely manner, it is also possible that individuals are doing necessary work not accounted for in plan. Other possibilities are that not enough time is being put into the planning activities, or that the on-going planning work is more involved than anticipated.

4.2.4 Impact and Corrective Action

The most likely impact of planning issues is that resources are diverted from other planned work in other areas of the project. It is rare that these major milestones or POP exercises would be delayed, it is more common that other things are put aside.

The corrective actions depend on the results of the analysis. If the planning work is understaffed, or if it is more complex than anticipated, resources need to be added. Another corrective action to consider is to make sure that additional management effort is explicitly planned in the month preceding a POP exercise or a major milestone.

4.3 SOFTWARE FUNCTIONALITY PROCEDURE

NOTE: As margins are reported at milestone reviews, they are deferred to a later version of this standard.

4.3.1 Objectives

There are two objectives for software functionality:

- Deliver the required software functionality
- Ensure system performance measures are within established margins

Both of these address measuring the product to evaluate whether the system delivered for operations will meet all its requirements and perform within the planned resource margins.

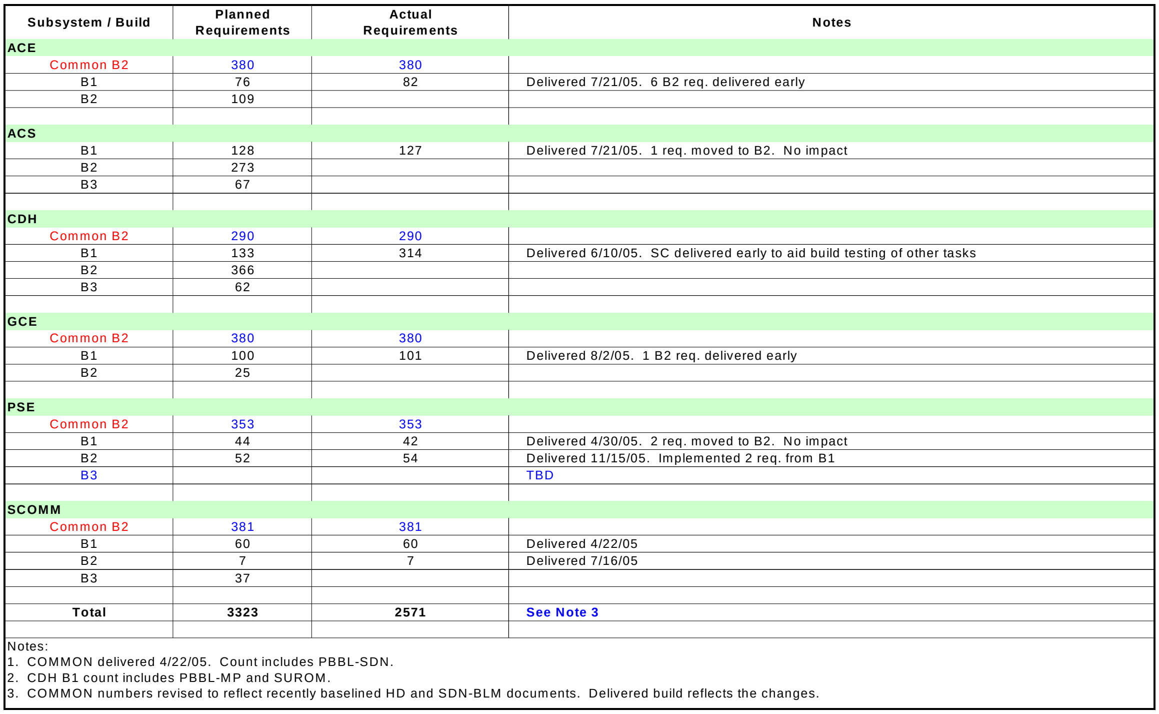
4.3.2 Measures

The measures used to analyze software functionality are:

- For each build of each subsystem, the number of requirements planned and the number of requirements delivered.

For the purpose of counting requirements, the builds measured are those delivered to the test team; builds internal to the development team are excluded. These data are reported monthly at Branch Status Reviews.

These data are collected in the FSW Metrics Spreadsheet, under the “RD Data” tab. Figure 4.3-1 shows a portion of this worksheet.



Subsystem / Build	Planned Requirements	Actual Requirements	Notes
ACE			
Common B2	380	380	
B1	76	82	Delivered 7/21/05. 6 B2 req. delivered early
B2	109		
ACS			
B1	128	127	Delivered 7/21/05. 1 req. moved to B2. No impact
B2	273		
B3	67		
CDH			
Common B2	290	290	
B1	133	314	Delivered 6/10/05. SC delivered early to aid build testing of other tasks
B2	366		
B3	62		
GCE			
Common B2	380	380	
B1	100	101	Delivered 8/2/05. 1 B2 req. delivered early
B2	25		
PSE			
Common B2	353	353	
B1	44	42	Delivered 4/30/05. 2 req. moved to B2. No impact
B2	52	54	Delivered 11/15/05. Implemented 2 req. from B1
B3			TBD
SCOMM			
Common B2	381	381	
B1	60	60	Delivered 4/22/05
B2	7	7	Delivered 7/16/05
B3	37		
Total	**3323**	**2571**	**See Note 3**

Notes:
1. COMMON delivered 4/22/05. Count includes PBBL-SDN.
2. CDH B1 count includes PBBL-MP and SUROM.
3. COMMON numbers revised to reflect recently baselined HD and SDN-BLM documents. Delivered build reflects the changes.

Figure 4.3-1 Requirements Data by Build

4.3.3 Analysis

As this data is a snapshot of delivery status, rather than monthly trending data, it can be analyzed by directly comparing the number of planned requirements and the number actually delivered. The text below defines analysis and reporting for the cases when the actual number of requirements is greater than, equal to , or less than the number planned.

Greater Report reason at BSR and take credit for being ahead. Possible reasons can include configuring common code in a build prior to when testers will be addressing the requirement, or taking initiative to solve a risky problem earlier than a scheduled build.

Equal Report that everything is working as planned.

Less If the actual delivery is less than 90% of requirements, report on why this occurred and raise the issue as a heightened risk if warranted. Some potential causes for not delivering all planned requirements are: more complexity, collaborator (e.g. HW people) requires early delivery, waiting on delivery from collaborator (e.g. algorithms from 590), understaffing, low productivity.

If more than 90% of requirements are delivered, but one or more is critical, report as heightened risk, and address what is being done to correct the issue.

Otherwise, there is a good chance to make up ground on the next builds, unless this report is describing one of the last builds.

4.3.4 Impact & Corrective Action

The impact of delivering fewer requirements may be added cost to the project. To determine this, one should cross check with the progress point counting for the affected subsystems. If the progress is within bounds, there may not be a significant impact yet, but the risk should reported and tracked. If progress is also lagging, the options are a) add staff to keep on schedule, b) extend schedule, or for early builds, raise the risk level and consider how to address the risk if the lack of progress persists.

When raising the level of risk reported, one considers how difficult the missing requirements are to implement, and whether the reporting is for an early or late build. The likelihood and impact set for the risk is decided through engineering judgment.

4.4 SOFTWARE QUALITY PROCEDURE

4.4.1 Objective

The objective for software quality is to ensure that system delivered for operations has no critical or moderate severity errors.

4.4.2 Measures

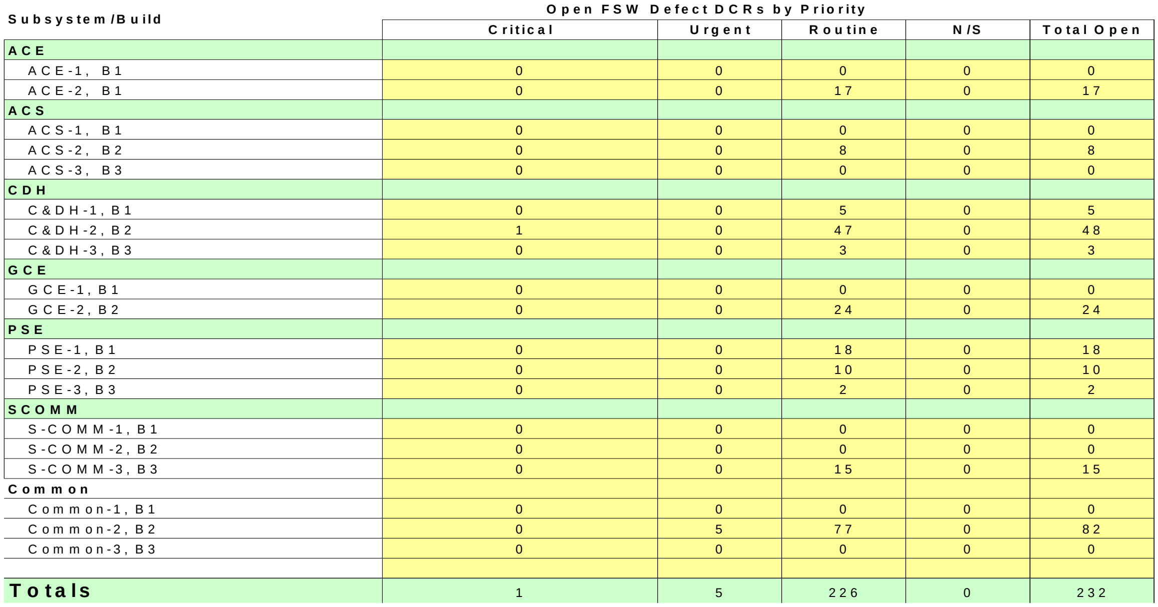
The measures used to assess software quality are:

- Open and closed defects by severity for each subsystem and build
- Distribution of defects by product area; both FSW and non-FSW entities (e.g. flight software, ground system, simulator, hardware…).

The defects considered for these measures are those reported in DCRs against delivered code. They do not include defects reported from inspections or walkthroughs; the latter are analyzed as part of the Verification procedure. All of this data is reported at monthly Branch Status Reviews. The data are imported from the MKS tool into two separate tabs; one contains data for DCRs related to flight software defects, the second contains data for defect DCRs related to products other than the software itself.

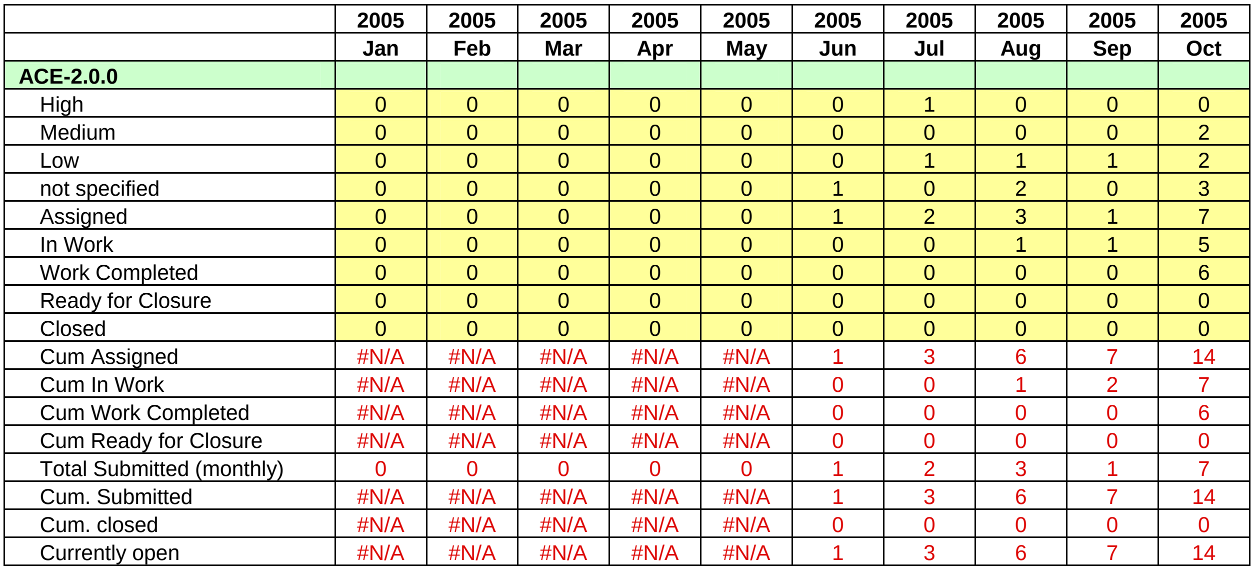
The data from the two imported lists automatically generates two more tabs; one presents the defects with respect to builds, and the other with respect to date. They are found in the FSW Metrics Spreadsheet under the “DCR Data – Build” and “DCR Data – Date”, respectively. Figures 4.4-1 and 4.4-2 show slices of these tables.

Figure 4.4-1 identifies DCRs where the priority is not specified (“N/S” in Figure 4.4-1); this indicates erroneous input data, as all defects should have a priority associated with them. One should see nothing but zeros for the unspecified priority.



Subsystem /Build	Open FSW Defect DCRs by Priority				
	Critical	Urgent	Routine	N/S	Total Open
ACE					
ACE-1, B1	0	0	0	0	0
ACE-2, B1	0	0	17	0	17
ACS					
ACS-1, B1	0	0	0	0	0
ACS-2, B2	0	0	8	0	8
ACS-3, B3	0	0	0	0	0
CDH					
C&DH-1, B1	0	0	5	0	5
C&DH-2, B2	1	0	47	0	48
C&DH-3, B3	0	0	3	0	3
GCE					
GCE-1, B1	0	0	0	0	0
GCE-2, B2	0	0	24	0	24
PSE					
PSE-1, B1	0	0	18	0	18
PSE-2, B2	0	0	10	0	10
PSE-3, B3	0	0	2	0	2
SCOMM					
S-COMM-1, B1	0	0	0	0	0
S-COMM-2, B2	0	0	0	0	0
S-COMM-3, B3	0	0	15	0	15
Common					
Common-1, B1	0	0	0	0	0
Common-2, B2	0	5	77	0	82
Common-3, B3	0	0	0	0	0
Totals	1	5	226	0	232

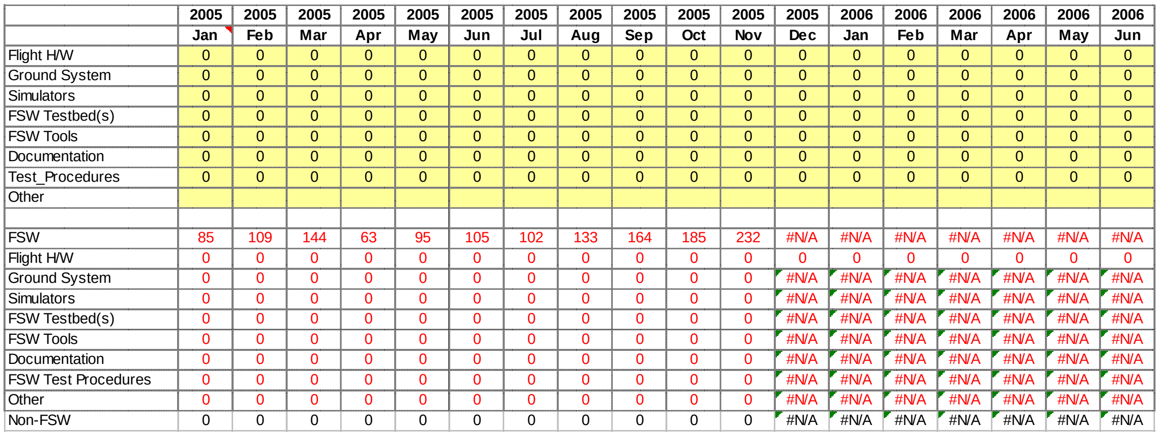
Figure 4.4-1 DCR Data by Build



	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
ACE-2.0.0										
High	0	0	0	0	0	0	1	0	0	0
Medium	0	0	0	0	0	0	0	0	0	2
Low	0	0	0	0	0	0	1	1	1	2
not specified	0	0	0	0	0	1	0	2	0	3
Assigned	0	0	0	0	0	1	2	3	1	7
In Work	0	0	0	0	0	0	0	1	1	5
Work Completed	0	0	0	0	0	0	0	0	0	6
Ready for Closure	0	0	0	0	0	0	0	0	0	0
Closed	0	0	0	0	0	0	0	0	0	0
Cum Assigned	#N/A	#N/A	#N/A	#N/A	#N/A	1	3	6	7	14
Cum In Work	#N/A	#N/A	#N/A	#N/A	#N/A	0	0	1	2	7
Cum Work Completed	#N/A	#N/A	#N/A	#N/A	#N/A	0	0	0	0	6
Cum Ready for Closure	#N/A	#N/A	#N/A	#N/A	#N/A	0	0	0	0	0
Total Submitted (monthly)	0	0	0	0	0	1	2	3	1	7
Cum. Submitted	#N/A	#N/A	#N/A	#N/A	#N/A	1	3	6	7	14
Cum. closed	#N/A	#N/A	#N/A	#N/A	#N/A	0	0	0	0	0
Currently open	#N/A	#N/A	#N/A	#N/A	#N/A	1	3	6	7	14

Figure 4.4-2 DCR Data by Date

Finally, the data for Dependencies on Non-FSW Defect DCRs can be found in the FSW Metrics Spreadsheet under the "DCR Data – Flavor" tab. These data are also generated from the two imported tabs. Figure 4.4-3 shows a portion of this worksheet, which is also generated from the imported data.

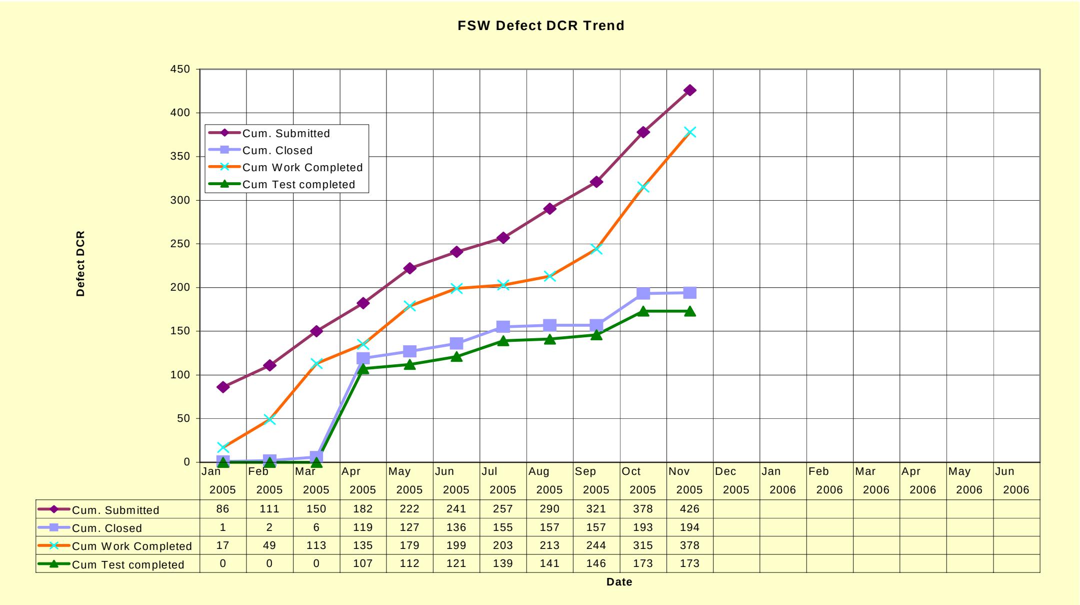


	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2006	2006	2006	2006	2006	2006
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
Flight H/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ground System	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Simulators	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FSW Testbed(s)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FSW Tools	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Documentation	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Test_Procedures	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Other																		
FSW	85	109	144	63	95	105	102	133	164	185	232	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Flight H/W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ground System	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Simulators	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
FSW Testbed(s)	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
FSW Tools	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Documentation	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
FSW Test Procedures	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Other	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Non-FSW	0	0	0	0	0	0	0	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A

Figure 4.4-3 Dependencies on Non-FSW Defect DCRs (table)

4.4.3 Analysis

The defect data can be examined for the flight software as a whole, or for each individual build. The overall chart is shown in Figure 4.4-4; the charts for each subsystem are similar. These charts look at how defect opening and closing is evolving over time, and answers the question of whether defects are being resolved quickly enough.



	Jan 2005	Feb 2005	Mar 2005	Apr 2005	May 2005	Jun 2005	Jul 2005	Aug 2005	Sep 2005	Oct 2005	Nov 2005	Dec 2005	Jan 2006	Feb 2006	Mar 2006	Apr 2006	May 2006	Jun 2006
Cum. Submitted	86	111	150	182	222	241	257	290	321	378	426							
Cum. Closed	1	2	6	119	127	136	155	157	157	193	194							
Cum Work Completed	17	49	113	135	179	199	203	213	244	315	378							
Cum Test completed	0	0	0	107	112	121	139	141	146	173	173							

Analysis: Nearly 90% of the submitted DCRs have been corrected by the development team. Test Completion and Closure rate lagging a bit, but gap expected to close as more of the "Work Completed" DCRs go to the Test Team.
Impact: No impact.
Corrective Action: None needed.

Figure 4.4-4. Defect Data Trend Plot.

The desirable behavior of this plot is to have the number of DCRs closed converge to meet the number of DCRs submitted. The analysis of these data depends on where the project is in its timeline. If the project is near completion and the curve is not converging, staff will need to be shifted to work on closing these DCRs. Earlier in the project or in a build more DCRs are being submitted and the expectation is that the closed curve is not diverging from the total curve. There is a lag between when DCRs start accumulating and when the first changes are completed by developers, after this point diverging curves should be taken as a warning sign.

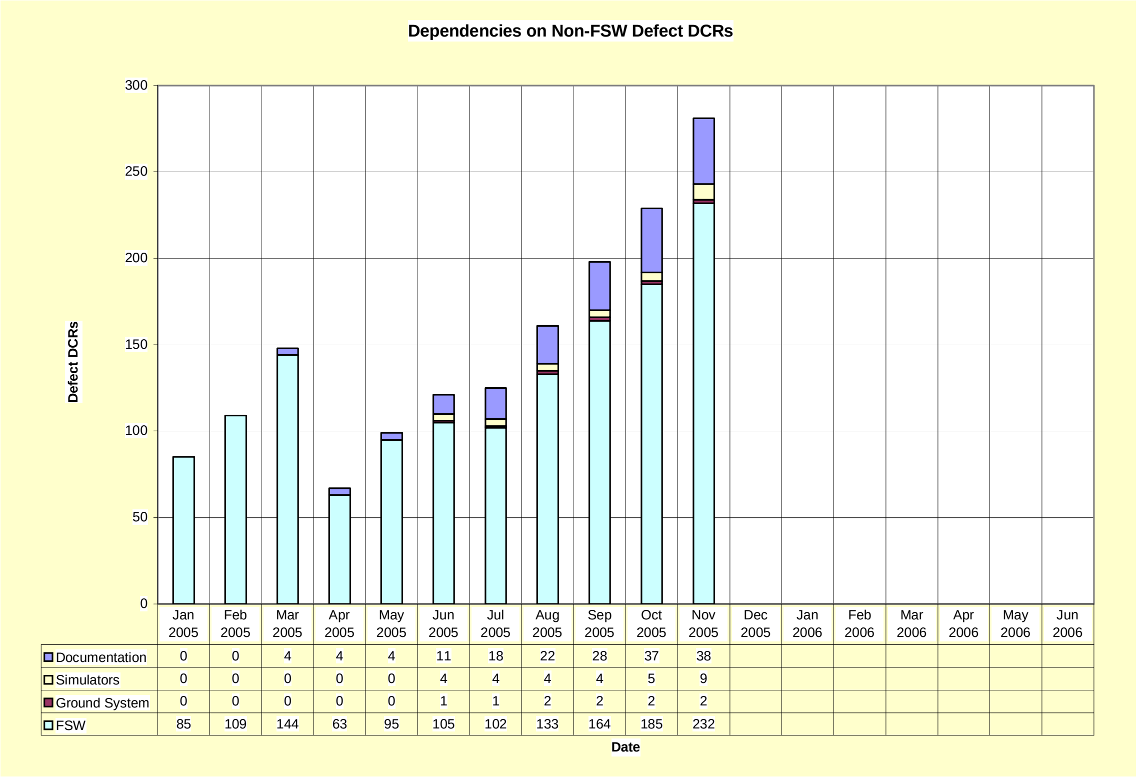
If the DCRs are being closed at a sufficient rate, report this on the BSR slide showing this plot. If not, shift resources to help close them faster, and report the schedule impact of this shift.

Examining these trend graphs is the main analysis done for defects, but examining the data by build and by product area can be used to locate problem areas in the software.

Figure 4.4-5 presents DCR data organized by FSW dependency. The intent of this chart is to analyze the positive or negative effects of non-FSW entities on the FSW team.

The graph shows the number of defects in several different areas; one wants to see the number of defects in each area stay under control early in the project, and converge towards zero as it moves onward. The PDL defines "under control" using his or her engineering judgment; and this analysis is reviewed at BSR meetings.

Beyond the general trends, the analysis needs to highlight significant dependency issues. These can be hindrances such as delayed testbed delivery, or positive events such as delivering a complete, clearly written algorithm document. In both cases the impact should be noted. For delayed deliveries the impact is the obvious cost or schedule effect; for positive events one can express the impact as a reduction of risk due to typical problems being less likely to occur.



	Jan 2005	Feb 2005	Mar 2005	Apr 2005	May 2005	Jun 2005	Jul 2005	Aug 2005	Sep 2005	Oct 2005	Nov 2005	Dec 2005	Jan 2006	Feb 2006	Mar 2006	Apr 2006	May 2006	Jun 2006
■ Documentation	0	0	4	4	4	11	18	22	28	37	38							
□ Simulators	0	0	0	0	0	4	4	4	4	5	9							
■ Ground System	0	0	0	0	0	1	1	2	2	2	2							
□ FSW	85	109	144	63	95	105	102	133	164	185	232							

Figure 4.4-5 Dependencies on Non-FSW Defect DCRs (chart)

Any corrective action needs to be planned in conjunction with the external group that the FSW team is counting on. If a meeting of minds can't be reached the issue needs to be made visible to FSB management, and possibly to Project management as well.

The build data are analyzed to locate particular areas that need attention to defect correction. In this case one examines the defect by build data (Figure 4.4-1) to determine if the number of defects is excessive. This is an engineering judgment that is made by weighing the number of defects against the size and complexity of the build. If an experienced development team lead or product development lead comes to the judgment that a particular build has too many defects, at the very least he or she should investigate why, and, if investigation warrants, shift resources to correcting these defects.

4.4.4 Impact and Corrective Action

If defects aren't closing fast enough, the impact is either delay of build completion or the inability to close all the DCRs before build delivery.

Some available corrective actions are to:

- Shift some of the staff to working on defects to meet schedule, on the assumption that it can be addressed from contingency funds or by making up ground later.

- Defer less critical DCRs to future builds or to the maintenance team. If this is done, it should be monitored as a risk.

- If the number of DCRs is extremely large in the team's judgment, one should plan to increase budget to address them, or to descope other work to accommodate this added effort.

4.5 CONFIGURATION MANAGEMENT PROCEDURE

4.5.1 Objectives

The objectives for Configuration Management are to ensure:

- That configuration management is being performed as planned.

- Adequacy of resources for configuration management.

4.5.2 Measures

The measures for configuration management include:

- Number of IRB and VDD Review meetings vs. plan.

- Planned and actual effort spent on CM.

The effort data is generated automatically in the Staffing Spreadsheet, on the "Effort by Process" tab. This tab is copied to the Status Spreadsheet, where the analysis is performed.

The counts of IRB and VDD meetings are found in the FSW Metrics Spreadsheet, under the "Proc. Data" tab (see Figure 4-1).

4.5.3 Analysis

Nominally, all the planned meetings will be held and the actual effort will be within 25% of the planned effort for the month. If the actual number of either type of meeting falls short of the plan, it is considered outside the nominal range.

If the actual effort is more than 25% away from the plan for a month, examine past data to determine if this is a persistent issue; if it is above the planned amount look further to see if there is configuration management work not accounted for in your planned effort. If no corrective action is needed, simply report the reason for the one-time deviation.

If the actual effort is below this upper bound, but the meetings are not being held on schedule, it is possible that configuration management is understaffed, or key personnel are diverted to other tasks. For the CMO, other tasks may be high-priority support for another project. Other personnel such as the PDL may have higher priority issues to resolve – it is OK to report meetings being deferred to fight fires.

In addition to measuring whether or not meetings occur, these measures are indicators that all the work leading up to being able to hold a meeting has occurred. Part of the analysis may be to identify what work needs to be completed that hasn't been.

Another possible cause of CM problems is a situation where the mission is more complex than the original staffing plan anticipated, affected the effort needed for CM.

4.5.4 Impact and Corrective Action

The impact of missing CM activities is an increased risk of version control problems, or of omitting required products or components from a delivery. In the latter case, it is more likely that documentation or other supporting products, rather than the flight software itself, will be omitted.

The corrective action is typically to add staff to assure that all necessary work is being completed. A second possibility is increased monitoring by the PDL to assure that all the activities leading up to IRB and VDD Review meetings are being carried out in a timely manner.

If the behavior is nominal, or if deviation from nominal is a one-time, unlikely to repeat anomaly, then no corrective action needs to be reported. However, the analysis needs to clearly state how the data is behaving.

4.6 REQUIREMENTS MANAGEMENT PROCEDURE

4.6.1 Objectives

There are three objectives for requirements management:

- Ensure requirements are being managed as planned.
- Ensure requirements are complete and stable enough to continue work without undue risk.
- Ensure adequacy of resources for requirements management.

The first of these is a measure of whether the requirements management processes are being carried out as planned, and is used as evidence for the Requirements Management Process Area of CMMI. The second is to examine requirements with an eye to whether they pose a risk to future progress.

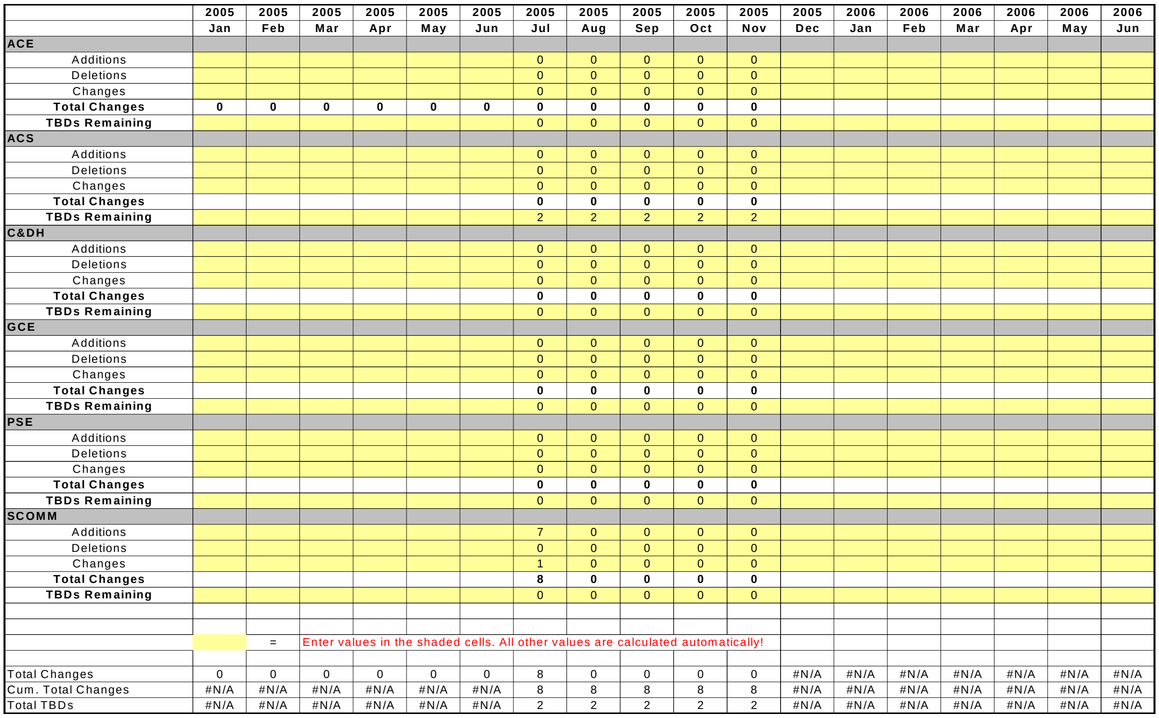
4.6.2 Measures

The measures used to analyze requirements management are:

- Requirements changes (additions, deletions and changes) by subsystem and total for all subsystems.
- Requirements TBDs by subsystem and total for all subsystems.
- Planned and actual effort spent on engineering.

The first two measures are collected in the FSW Metrics Spreadsheet, under the "RV Data" tab. RV indicates "requirements volatility", the jargon used by NPR 7150.2. Figure 4.6-1 below shows a portion of this worksheet. Both of these measures apply once requirements are placed under change control by the team's Configuration Management Officer, and are reported at a monthly BSR.

The effort data is generated automatically in the Staffing Spreadsheet, on the "Effort by Process" tab. This tab is copied to the Status Spreadsheet, where the analysis is performed.



	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2005	2006	2006	2006	2006	2006	2006
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
ACE																		
Additions							0	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							0	0	0	0	0							
Total Changes	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**							
TBDs Remaining							0	0	0	0	0							
ACS																		
Additions							0	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							0	0	0	0	0							
Total Changes							**0**	**0**	**0**	**0**	**0**							
TBDs Remaining							2	2	2	2	2							
C&DH																		
Additions							0	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							0	0	0	0	0							
Total Changes							**0**	**0**	**0**	**0**	**0**							
TBDs Remaining							0	0	0	0	0							
GCE																		
Additions							0	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							0	0	0	0	0							
Total Changes							**0**	**0**	**0**	**0**	**0**							
TBDs Remaining							0	0	0	0	0							
PSE																		
Additions							0	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							0	0	0	0	0							
Total Changes							**0**	**0**	**0**	**0**	**0**							
TBDs Remaining							0	0	0	0	0							
SCOMM																		
Additions							7	0	0	0	0							
Deletions							0	0	0	0	0							
Changes							1	0	0	0	0							
Total Changes							**8**	**0**	**0**	**0**	**0**							
TBDs Remaining							0	0	0	0	0							
	🟨	=	Enter values in the shaded cells. All other values are calculated automatically!															
Total Changes	0	0	0	0	0	0	8	0	0	0	0	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Cum. Total Changes	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	8	8	8	8	8	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A
Total TBDs	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	2	2	2	2	2	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A	#N/A

Figure 4.6-1 Requirements Volatility Data

4.6.3 Analysis

Figure 4.6-2 shows the plot generated from the data in Figure 4.6-1. It is found under the "RV Plot" tab in the FSW Metrics Spreadsheet. This plot shows the cumulative growth of the number of requirements changes in purple, and the (one hopes) declining number of TBDs in light blue.

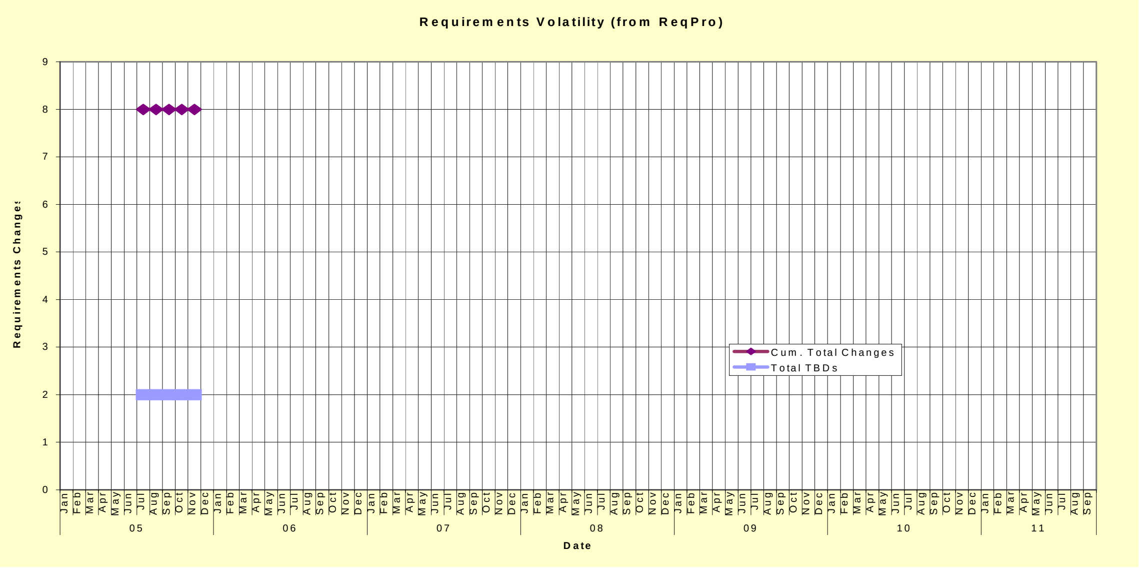


Figure 4.6-2 Requirements Volatility Plot

The first, and more important, analysis is to determine whether the requirements are complete enough and stable enough to proceed with development. In reality, one often proceeds with development even to unstable or incomplete requirements, the purpose of this measurement is to identify when this is happening and understand that risk is being heightened.

Unlike the effort, progress and schedule analysis in the Project Monitoring and Control procedure, there is no specific numeric threshold defining when a problem is occurring. This analysis is done by looking at the shape of the curves and at where the project is in its timeline. One would expect TBD requirements to have little impact by CDR time, and similarly one would expect the number of requirements changes to diminish as the project progresses.

If in the team’s judgment this behavior this is not happening well enough, a corresponding risk needs to be added or an existing risk modified to increase the risk exposure. This increased risk should be reported at BSR if the engineering judgment is that the exposure is medium or high. If the team is satisfied with the behavior of this data, it simply needs to report this on the BSR slide.

The same data that is used to assess requirements volatility can be used to determine if the Requirements Management process area is being properly addressed. To do this, examine the number of requirements changes and of TBDs to see if these values are changing. If you see identical values in two or more consecutive months, you should be suspicious. Depending on the situation, you may be waiting for someone else to resolve TBDs, in which case your process is just fine, or the lack of changes may reflect inadequate attention to keeping requirements and the associated traceability matrices up to date.

If the actual effort spent on engineering is more than 25% away from the plan for a month, examine past data to determine if this is a persistent issue; if it is above the planned amount look further to see if there is requirements management work not accounted for in your planned effort. If no corrective action is needed, simply report the reason for the one-time deviation.

4.6.4 Impact and Corrective Action

The impact of excessive requirement changes or persistent TBDs can be delay, or possible quality problems with software delivered to test teams. Ultimately, this may translate into added cost or delays.

To determine if there is such an impact, one can examine the detailed data on the RV data tab. If a high number of changes or TBDs correlates with lack of progress for a given subsystem, a possible corrective action is to add or shift personnel (the right personnel!) to address the requirements issues before they get out of hand. Another indicator that more effort needs to be put into requirements management and development work is actual effort for engineering activities consistently running behind plan.

If there are no strong cross correlations with progress, but TBD or change issues are persisting (particularly with critical requirements where a resolution date can be established), this issue should be added to the risk list and monitored closely.

4.7 PROCESS AND PRODUCT QUALITY ASSURANCE PROCEDURE

4.7.1 Objectives

The objectives for Process and Product Quality Assurance are to:

- Ensure software assurance is being performed as planned.
- Ensure adequacy of resources for process and product QA

4.7.2 Measures

The measures for PPQA are:

- Number of evaluations (planned and actual)
- Planned and actual effort spent on QA.

The effort data is generated automatically in the Staffing Spreadsheet, on the “Effort by Process” tab. This tab is copied to the Status Spreadsheet, where the analysis is performed (see Figure 4.1-11).

The remaining data are collected in the FSW Metrics Spreadsheet, under the “Proc. Data” tab (see Figure 4-1).

4.7.3 Analysis

Nominally, all the planned evaluations are held and the actual effort will be within 25% of the plan. If the number of evaluations is short of planned, then it is considered to be out of the nominal range. This is somewhat unlikely, as these evaluations are far less frequent than the monthly BSR.

If the actual effort is more than 25% away from the plan for a month, examine past data to determine if this is a persistent issue; if it is above the planned amount look further to see if there is process and product quality assurance work not accounted for in your planned effort. If no corrective action is needed, simply report the reason for the one-time deviation.

If the actual effort is consistently below the upper bound, and if the work is being done on schedule, then either the work is easier than anticipated or is being performed more efficiently than anticipated. If the work is not on schedule, it is possible that process and product quality assurance is understaffed. If the latter is the case, one will need to work with Code 300 to assure that there are adequate SQE resources available for these evaluations.

4.7.4 Impact and Corrective Action

The purpose of process and product quality assurance is to assure that processes are being carried out as documented, and that products conform to all applicable standards. Thus, beyond budget and schedule impact, the impact of missing activities is risk may be added through not being aware of deviations from standard, well-tested practices.

The corrective action is typically to work with Code 300 to assure proper staffing, and to assure coordination of their scheduled activities with software team activities.

If the behavior is nominal, or if deviation from nominal is a one-time, unlikely to repeat anomaly, then no corrective action needs to be reported. However, the analysis needs to clearly state how the data is behaving.

4.8 MEASUREMENT AND ANALYSIS PROCEDURE

4.8.1 Objectives

The objectives for measurement and analysis are to:

- Ensure project measures are collected, stored analyzed, and reported as planned.
- Ensure adequacy of resources for measurement and analysis.

4.8.2 Measures

The measures used to analyze the Measurement and Analysis Process Area are:

- Planned and actual effort spent on measurement for the month.
- Planned and actual number of completed monthly measurement spreadsheets added to repository.
- Planned and actual number of monthly BSR meetings.

The effort data is generated automatically in the Staffing Spreadsheet, on the “Effort by Process” tab. This tab is copied to the Status Spreadsheet, where the analysis is performed (see Figure 4.1-11).

The remaining data are collected in the FSW Metrics Spreadsheet, under the “Proc. Data” tab (see Figure 4-1).

4.8.3 Analysis

Nominally, all the planned data are collected, all the planned BSR meetings are held, and the actual effort will be within 25% of the plan. If either the number of spreadsheets collected or the number of BSRs supported are short of planned number, then they are considered to be out of the nominal range.

If the actual effort is more than 25% away from the plan for a month, examine past data to determine if this is a persistent issue; if it is above the planned amount look further to see if there is measurement and analysis work not accounted for in your planned effort. If no corrective action is needed, simply report the reason for the one-time deviation.

If the actual effort is consistently below the upper bound, and if the work is being done on schedule, then either the work is easier than anticipated or measurement and analysis is being performed more efficiently than anticipated. If the work is not on schedule, it is possible that measurement and analysis is understaffed, or that the PDL and team leads who add their analysis for the BSR are diverted to higher priority issues. If the latter is the case, one should report the work being deferred to fight fires.

In addition to measuring whether or not spreadsheets are collected and BSRs supported, these measures are indicators that all the measurement collection, storage, and analysis leading up to being able to hold a BSR has occurred. Part of the analysis for measurement and analysis may be to identify what work needs to be completed that hasn't been.

Another possible cause of measurement and analysis problems is a situation where the mission is more complex than the original staffing plan anticipated, affected the effort needed for measurement and analysis.

4.8.4 Impact and Corrective Action

The main point of measurement and analysis activities is to have timely, quantitative assessments of a project's health. Thus, beyond budget and schedule impact, the impact of missing measurement and analysis activities is that it may take longer to detect issues that put a project to risk.

The corrective action is typically to add staff to assure that all necessary work is being completed. A second possibility is increased monitoring by the PDL to assure that all the activities leading up to a BSR are being carried out in a timely manner.

If the behavior is nominal, or if deviation from nominal is a one-time, unlikely to repeat anomaly, then no corrective action needs to be reported. However, the analysis needs to clearly state how the data is behaving.

4.9 VERIFICATION PROCEDURE

Deferred to next version.

4.9.1 Objectives

4.9.2 Measures

4.9.3 Analysis

4.9.4 Impact and Corrective Action

4.10 VALIDATION PROCEDURE

Deferred to next version.

4.10.1 Objectives

4.10.2 Measures

4.10.3 Analysis

4.10.4 Impact and Corrective Action